Particle Competition and Cooperation to Prevent Error Propagation from Mislabeled Data in Semi-Supervised Learning

Fabricio Breve*
Institute of Geosciences and Exact Sciences (IGCE)
São Paulo State University (UNESP)
Rio Claro, Brazil
Email: fabricio@rc.unesp.br

Liang Zhao
Institute of Mathematics and Computer Science (ICMC)
University of São Paulo (USP)
São Carlos, Brazil
Email: zhao@icmc.usp.br

***Abstract*—Semi-supervised learning is applied to classification problems where only a small portion of the data items is labeled. In these cases, the reliability of the labels is a crucial factor, because mislabeled items may propagate wrong labels to a large portion or even the entire data set. This paper aims to address this problem by presenting a graph-based (network-based) semi-supervised learning method, specifically designed to handle data sets with mislabeled samples. The method uses teams of walking particles, with competitive and cooperative behavior, for label propagation in the network constructed from the input data set. The proposed model is nature-inspired and it incorporates some features to make it robust to a considerable amount of mislabeled data items. Computer simulations show the performance of the method in the presence of different percentage of mislabeled data, in networks of different sizes and average node degree. Importantly, these simulations reveals the existence of the critical points of the mislabeled subset size, below which the network is free of wrong label contamination, but above which the mislabeled samples start to propagate their labels to the rest of the network. Moreover, numerical comparisons have been made among the proposed method and other representative graph-based semi-supervised learning methods using both artificial and real-world data sets. Interestingly, the proposed method has increasing better performance than the others as the percentage of mislabeled samples is getting larger.**

***Keywords*-Machine learning; Computational intelligence**

I. Introduction

Semi-Supervised Learning is the name given to a group of machine learning algorithms specifically design to handle problems in which the data sets are composed by only a small subset of labeled samples, while the remaining samples are left unlabeled. This is a common situation nowadays, as the size of the data sets being treated is constantly increasing, making expensive the task of labeling enough samples for the training process, which usually requires the work of human experts [1]–[3].

The quality of the training data is crucial in semi-supervised learning due to the small amount of labeled data. Errors (wrong labels) may easily be propagated to a portion of or the entire data set. Humans and other animals can easily compensate for imperfect data in their learning process. Behavioral experiments show that animals can successfully learn from conditioning even when they are inconsistently rewarded. The same does not apply to machine learning systems, in which fault-tolerance is usually hard to achieve. Most algorithms just assume that the input label information is completely reliable, but in practice mislabeled samples are commonly found in the data sets. This problem is commonly refereed as *learning from imperfect data* [4], [5]. Though this is an important topic, it has not received much attention from researchers and there are still few works to study it in the semi-supervised learning domain [6]–[8].

Recently, Quiles et al. proposed a biologically inspired clustering algorithm using particle walking and competition to detect communities in networks [9]. Particles compete with each other in order to possess nodes of the network, naturally confining themselves within a cluster. Later, Breve et al. extended this approach to realize semi-supervised learning [10]. They used teams of particles which cooperate with their teammates and compete against particles of other teams. Particles are created for each labeled node of the network, and they try to conquer and defend their neighborhood. Those particle walking based methods provide good classification results compared to the well known methods, but with lower order of computational complexity. This approach naturally has some tolerance to some mislabeled data, as shown in [11], although it was not designed to handle this specific problem.

In this paper, we present a new semi-supervised learning particle walking algorithm which is specifically designed to handle data sets with large amounts of mislabeled data. It is based on the model in [10]. The main feature of the proposed model is the improvement in the robustness of the label propagation process in the presence of mislabeled samples. Another contribution of this study is the uncovering of the critical points of the mislabeled subset size, below

*F. Breve is also with the Institute of Mathematics and Computer Science (ICMC), University of São Paulo (USP), São Carlos, Brazil, Email: fabricio@icmc.usp.br

This work was supported by the São Paulo State Research Foundation (FAPESP), the Brazilian National Research Council (CNPq), and the Foundation for the Development of Unesp (Fundunesp).

which the network is free of wrong label contamination, but above which the mislabeled samples start to propagate their labels to the rest of the network. Some other relevant features introduced in this paper are: labeled nodes that have the same label are all interconnected independently of their distance; there is only a distance table for each team, which is shared by all teammates; and all node potentials are variables, even those from labeled nodes. These features allow particles to leave mislabeled nodes, which will be usually inside other class neighborhood, and help their teammates in the neighborhood of their respective class. In our experiments, when there are mislabeled samples in the data sets, the proposed method outperforms the particles competition and cooperation method [10] and other representative graph-based semi-supervised learning methods.

Computer simulations presented in this paper show the effectiveness and robustness of the proposed method in the presence of mislabeled data. It also includes a study on how the performance of the algorithm is affected as some network attributes changes, like the network size and the average node degree. Finally, there is a comparison among the proposed method and other representative graph-based semi-supervised learning methods applied to some artificial and real-world data sets with different amounts of mislabeled data.

This paper is organized as follows. The proposed model is described in Section II. In Section III, we present computer simulations and numerical analysis. Finally, in Section IV we draw some conclusions.

II. Model Description

In this section, we introduce the proposed semi-supervised learning method, which relies on particle competition and cooperation. A set of particles, each of them representing a labeled data item, are put in an unweighted network. A subset of particles representing nodes with the same label is called a *team*. These teams will compete with each other to possess nodes of the network. Each node has a vector to represent the domination level of each team on it. While teammates particles act cooperatively to possess the nodes of the network, particles belonging to different teams will compete with each other trying to avoid rivals to enter their territory. At each iteration of the algorithm, each particle will choose a neighbor node to visit. The chosen node is called *target node*, and the particle which chooses the target node will increase its team domination level on this node, at the same time that it will decrease other teams domination levels on it. Each particle also has a strength level, which lowers or raises according to its team domination level on the target node. Each team has a distance table, which is used to avoid the particles from leaving their neighborhood unprotected. The team distance table is a new feature introduced in this work, replacing the individual particle distance tables used in [10].

The network is built from a given data set $\chi = \{x_1, x_2, \ldots, x_l, x_{l+1}, \ldots, x_n\} \subset \mathbb{R}^m$, with the corresponding label set $L = \{1, 2, \ldots, c\}$. The first l points $x_i (i \leq l)$ are labeled as $y_i \in L$ and the remaining points $x_u (l < u \leq n)$ are left unlabeled, i.e, $y_u = \emptyset$. We define an undirected graph $\mathbf{G} = (\mathbf{V}, \mathbf{E})$. $\mathbf{V} = \{v_1, v_2, \ldots, v_n\}$ is the set of nodes, where each one v_i corresponds to a sample x_i, and $\mathbf{E}$ is the set of edges (v_i, v_j). Two nodes v_i and v_j are connected if v_j is among the k-nearest neighbors of v_i or vice-versa using Euclidean distance. Also, v_i and v_j are connected if they are both labeled nodes with the same label, i.e., $y_i = y_j$ and $\{y_i, y_j\} \in L$, which is a feature introduced in this new model. Otherwise, v_i and v_j are disconnected. For each network node $v_i \in \{v_1, v_2, \ldots, v_l\}$, corresponding to a labeled data point $x_i \in \{x_1, x_2, \ldots, x_l\}$, there is a particle $\rho_i \in \{\rho_1, \rho_2, \ldots, \rho_l\}$ which initial position is at v_i.

Each particle ρ_j holds a variable $\rho_j^{\omega}(t) \in [0, 1]$ corresponding to the particle strength, which indicates how much the particle can change nodes levels at time t. Teams have a variable $\rho_{\mathbf{j}}^{\mathbf{d}}(\mathbf{t})$ which is accessed by all the particles belonging to the team. It is a distance table, i.e., a vector $\rho_{\mathbf{j}}^{\mathbf{d}}(\mathbf{t}) = \{\rho_i^{d_1}(t), \rho_i^{d_2}(t), \ldots, \rho_i^{d_n}(t)\}$, where each element $\rho_j^{d_i}(t) \in [0 \quad n-1]$ holds the distance measured between the closest labeled node of its team and the node v_i.

Each node v_i has a vector variable $\mathbf{v_i^{\omega}(t)} = \{v_i^{\omega_1}(t), v_i^{\omega_2}(t), \ldots, v_i^{\omega_c}(t)\}$, where each element $v_i^{\omega_\ell}(t) \in [0, 1]$ corresponds to the domination level of team ℓ over node v_i. For each node, the sum of the domination levels is always constant, $\sum_{\ell=1}^{c} v_i^{\omega_\ell} = 1$.

The initial domination levels are set differently for nodes corresponding to labeled and unlabeled samples. Those corresponding to labeled samples begin fully dominated by the corresponding team. On the other hand, those corresponding to unlabeled samples have all teams domination levels set equally. Therefore, for each node v_i, the initial level of domination vector $\mathbf{v_i^{\omega}}$ is set as follows:

$$v_i^{\omega_\ell}(0) = \begin{cases} 1 & \text{if } y_i = \ell \\ 0 & \text{if } y_i \neq \ell \text{ and } y_i \in L \\ \frac{1}{c} & \text{if } y_i = \emptyset \end{cases} . \qquad (1)$$

Each particle ρ_j has its initial position set to its corresponding labeled node and its initial strength set to maximum, $\rho_j^{\omega}(0) = 1$. Particles start knowing no distances except for those to the labeled nodes of its own team, which are set to zero ($\rho_j^{d_i} = 0$). Other distances are set to the largest possible value ($\rho_j^{d_i} = n - 1$).

At each iteration t, each particle p_j selects a target neighbor node to visit. Each node v_i selected as a target, the domination level $v_i^{\omega_\ell}(t)$ is updated as follows:

$$v_i^{\omega_\ell}(t+1) = \begin{cases} \max\{0, v_i^{\omega_\ell}(t) - \frac{0.1\rho_j^{\omega}(t)}{c-1}\} \\ \quad \ell \neq \rho_j^f \\ v_i^{\omega_\ell}(t) + \sum_{q \neq \ell} v_i^{\omega_q}(t) - v_i^{\omega_q}(t+1) \\ \quad \ell = \rho_j^f \end{cases} , \qquad (2)$$

where ρ_j^f represents the class label of particle ρ_j. Each particle ρ_j will change the target node v_i by increasing the domination level of its team ($v_i^{\omega_\ell}$, $\ell = \rho_j^f$) while decreasing the domination levels of other teams ($v_i^{\omega_\ell}$, $\ell \neq \rho_j^f$)). Differently from [10], here labeled nodes domination levels are not fixed, which allows nodes to be relabeled if they were incorrectly labeled.

A particle will get weaker or stronger according to the domination level of its team in the target node. At each iteration, a particle strength is updated, $\rho_j^\omega(t) = v_i^{\omega_\ell}(t)$, where v_i is the target node, and $\ell = \rho_j^f$, i.e., ℓ is the class label of particle ρ_j.

Each particle ρ_j updates its team distance table $\rho_j^{d_k}(t)$ at each iteration t as follows:

$$\rho_j^{d_k}(t+1) = \begin{cases} \rho_j^{d_i}(t) + 1 & \text{if } \rho_j^{d_i}(t) + 1 < \rho_j^{d_k}(t) \\ \rho_j^{d_k}(t) & \text{otherwise} \end{cases}, \quad (3)$$

where $\rho_j^{d_i}(t)$ and $\rho_j^{d_k}(t)$ are the distances to the closest labeled node of its team from the current node and from the target node, respectively.

Each particle ρ_j chooses its target node v_i with probabilities defined according to its team domination level on that neighbor $\rho_j^{\omega_\ell}$ and the inverse of the distance ($\rho_j^{d_i}$) from that neighbor, v_i, to the closest labeled node of its team, v_j, as follows:

$$p(v_i|\rho_j) = 0.5 \left(\frac{W_{qi}}{\sum_{\mu=1}^{n} W_{q\mu}} + \frac{W_{qi} v_i^{\omega_\ell} \frac{1}{(1+\rho_j^{d_i})^2}}{\sum_{\mu=1}^{n} W_{q\mu} v_i^{\omega_\ell} \frac{1}{(1+\rho_j^{d_i})^2}} \right), \quad (4)$$

where q is the index of the node being visited by particle ρ_j and $\ell = \rho_j^f$, where ρ_j^f is the class label of particle ρ_j. A particle actually visits the target node only if its team domination level on that node is higher than those from all other teams; otherwise, a shock happens and the particle stays at the current node until the next iteration.

We monitor the average maximum domination levels of the nodes ($\langle v_i^{\omega^\ell} \rangle$, $\ell = \arg\max_q v_i^{\omega_q}$) and stop the algorithm when there is no increasing of this quantity. After the last iteration of the algorithm, each unlabeled node is labeled after the team which has the highest domination level on it, i.e., $y_i = \arg\max_\ell v_i^{\omega_\ell}(\infty)$.

III. Computer Simulations

In this section, we present simulation results to show the effectiveness and robustness of our method in the presence of mislabeled data. First, we evaluate the performance of the proposed method applied to networks with different sizes and average node degrees in the presence of mislabeled data. These artificial networks are generated using the method proposed by Danon et al. [12], with the nodes divided equally into 4 classes, and the classes mixture set to a moderate value, $z_{out}/\langle k \rangle = 0.25$, which means that approximately 25% of the edges are connecting different classes nodes.

For the first set of simulations, we generate networks with increasing number of nodes $n = \{64, 128, 192, 256, \ldots, 1024\}$, and average node degree proportional to the network size, $\langle k \rangle = n/8$. For each of these configurations we randomly select a subset of elements ($L \subset N$) to be labeled, while the others are presented to the algorithm without labels. The labeled subset size is set to $l/n = 0.1$ (10% labeled nodes is a typical semi-supervised learning problem). In order to test robustness to mislabeled samples, we randomly choose q elements from the labeled subset L ($Q \subset L$) to have their labels changed to any of the other classes chosen randomly for each sample, thus producing mislabeled nodes. These mislabeled subsets are generated with increasing sizes, $q/l = \{0.00, 0.05, 0.10, \ldots, 1.00\}$. So, we have 816 different configurations and each of them is repeated 100 times, with different generated networks and different samples in the labeled and in the mislabeled subsets, in order to obtain an average. The results are presented in Fig. 1a and by analyzing them we notice that as the network size increases ($n \to +\infty$), the performance curve with variable mislabeled samples subset size becomes rougher and the critical points, beyond which the performance drops, have a narrower angle.

In the second set of simulations, we generate networks with fixed size, $n = 512$. Networks are generated with 16 different levels of average node degree, $\langle k \rangle = \{8, 16, 24, 32, \ldots, 128\}$. The labeled subset is fixed, $l = 64$, and the mislabeled subset size is variable, $q/l = \{0.00, 0.02, 0.04, \ldots, 1.00\}$. Each of the 816 different configurations is repeated 100 times to obtain the averages. The results are presented in Fig. 1b, and by analyzing them we notice that as the average node degree increases from $\langle k \rangle = 8$ to $\langle k \rangle = 32$ there is an improvement in the algorithm performance. Moreover, the performance curve with variable mislabeled samples set size becomes smoother and the critical points have a wider angle. However, when the average node degree increases beyond $\langle k \rangle = 32$ there is no significatively differences in the algorithm performance.

By analyzing the shape of the graphics of Figs. 1a and 1b, we notice that, as the mislabeled subset grows, the algorithm performance is high and almost constant in the beginning, then there is a critical point where the correct classification rate starts to drop quickly, and finally there is another critical point where the correct classification rate begins to stabilize at a low value. These critical points vary with the parameters of the network, such as size and average node degree, as expected. We are not interested in the second critical point because, after this point, the quality of the labeled subset is worse than random labeling. In these cases, it would be better to use some unsupervised learning algorithm. The first critical point, on the other hand, is an important indicator of the robustness of the algorithm. The robustness of an algorithm in the presence of mislabeled samples may be

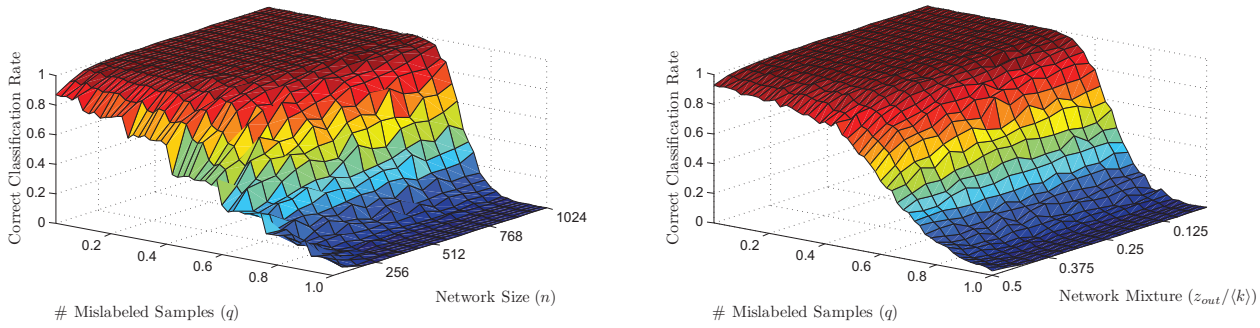

Figure 1. Correct Classification Rate with: (a) different network sizes and mislabeled subset sizes, $\langle k \rangle = n/8$, $l/n = 0.1$; and (b) different network mixtures and different average node degrees and mislabeled subset sizes. $n = 512$, $l = 64$.

measured by the size of the area before the first critical point, and also by the angle formed by the performance curve as the mislabeled subset increases. A large area before the first critical point followed by a sharp angle in the performance curve indicates that the algorithm is robust to mislabeled samples in that specific configuration. On the other hand, when the algorithm is not robust to mislabeled samples, the performance curve tends to a straight line and the critical points are hard to identify as they form wide angles. In our experiments, the larger area before the critical points and the critical points with sharper angles are obtained as the network size grows, which means the algorithm gets more robust in those cases. The area before the first critical point is quite large in most cases, which also points towards the robustness of the method. Regarding the network average node degree, as it increases from low to average, the area before the critical points becomes larger and the critical points get sharper angles, however, there is no point in increasing it too much as, beyond some level, there is no change in classification performance or location of critical points.

The performance of the algorithm in those typical semi-supervised learning setups is also impressive. In most cases, it managed to keep high correct classification rates even when there is a large percentage of mislabeled nodes. In Figs. 2a and 2b, we can observe the maximum size of the mislabeled subset that still produces good results (over 80% and 90% of correct classification rate) for different network sizes and average node degree.

Now we compare the proposed method with other representative graph-based semi-supervised learning methods using artificial and real-world networks with mislabeled data. In these experiments, the following techniques are considered: Local and Global Consistency (LGC) [13], Label Propagation (LP) [14], Linear Neighborhood Propagation (LNP) [15], and the original Particle Competition and Cooperation (PCC) method [10]. The σ parameters of the LGC and the LP methods, and the k parameters of LNP, PCC and the proposed method, are all optimized using the genetic algorithm available in the Global Optimization Toolbox of MATLAB. For the LGC and LNP methods, we have fixed $\alpha = 0.99$, as done in [13] and [15], respectively. For the PCC method, the following parameters are kept fixed: $p_{\text{grd}} = 0.5$, $\Delta_v = 0.1$, as done in [11].

We randomly select a subset of elements ($L \subset N$) to be labeled, while the others are presented to the algorithm without labels. The only exception is the Digit1 data set, where we use the same labeled subsets shown in [2]. In order to test robustness to mislabeled samples, we randomly choose q elements from the labeled subset L ($Q \subset L$) to have their labels changed to any of the other classes chosen randomly for each sample, thus producing mislabeled nodes. These mislabeled subsets are generated with increasing sizes, $q/l = \{0.00, 0.05, 0.10, \dots\}$. Each configuration is repeated at least 50 times, with different samples in the mislabeled subset, in order to obtain an average.

Figure 3 shows the performance comparison when the semi-supervised learning graph-based methods are applied to artificial data sets with $1,000$ elements equally divided into 4 normally distributed classes (Gaussian distribution). These data sets are generated by using function `gauss` from PRTools [16]. At each run, 50 samples are randomly chosen to compose the labeled subset. When all the samples in the labeled subset are correctly labeled, all the methods have similar performance. As the mislabeled subset increases, the classification error rate of all algorithms have a slightly increase in the beginning, followed by a larger increase at some point (the critical point), as expected. Notice that the proposed method is the last to reach its critical a point, i.e., it is more robust to mislabeled samples than the others. Its classification error rate is almost constant with up to 45% of mislabeled samples in the labeled subset, which is pretty impressive. The performance starts to decrease only when there are 50% or more mislabeled samples.

Figure 3b shows the performance comparison when the methods are applied to Digit1 data set [2]. The Digit1 data set is composed by 1500 samples divided into 2 classes, and it has 241 dimensions. There are 12 different labeled

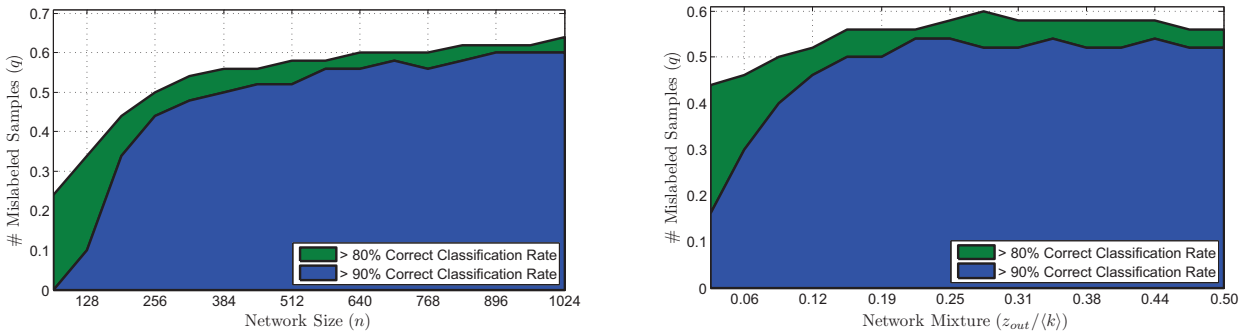

Figure 2. Maximum mislabeled subset size for 80% and 90% of correct classification rate with: (a) different network sizes, $\langle k \rangle = n/8$, $l/n = 0.1$; and (b) different network average node degree ($\langle k \rangle$), $n = 512$, $l/n = 0.1$.

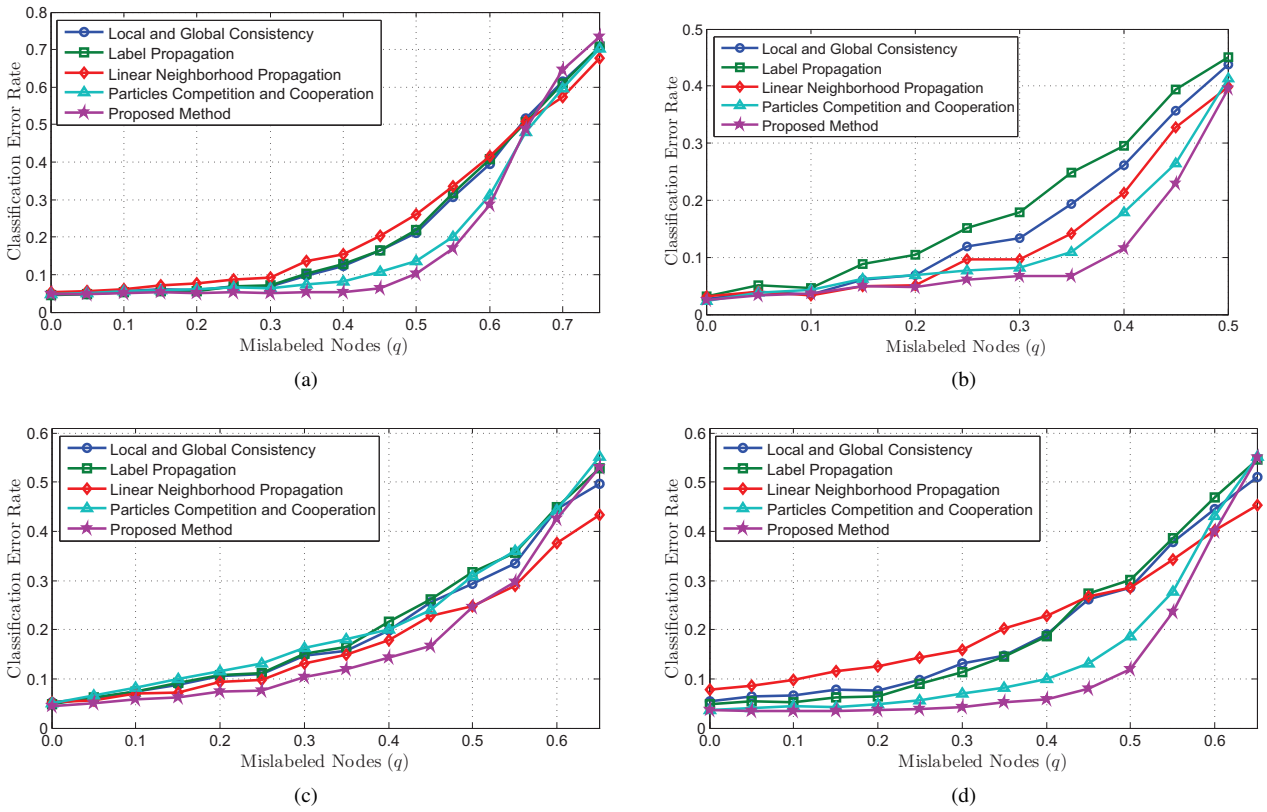

Figure 3. Classification error rate in: (a) a network with 4 normally distributed classes with different mislabeled subset size; (b) the Digit1 data set with different mislabeled subset size; (c) the Iris data set with different mislabeled subset size; (d) the Wine data set with different mislabeled subset size.

subset, each of them containing 100 samples, as provided in [2]. By analyzing Fig. 3b, we see that when all the samples in the labeled subset are correctly labeled, all the methods have good and similar performance. As the mislabeled subset increases, the classification error rate of all algorithms increases as well, and the advantage of the proposed method becomes visible, as it manages to have less performance degradation than the others, i.e., it is less affected by the mislabeled samples. In fact, with 35% of mislabeled samples, it has less than half of the amount of classification mistakes made by LGC, LP and LNP.

Figure 3c shows the performance comparison when the semi-supervised learning graph-based methods are applied to the Iris Data Set [17]. 40 samples are randomly chosen to compose the labeled subset. When all the samples in the labeled subset are correctly labeled, the proposed algorithm performs slightly better than the others. As the mislabeled subset increases, the performance of all algorithms decreases as expected, but the proposed method performance degrades less than the others, and its advantage becomes increasingly

more visible with up to 45% of the labeled subset composed by mislabeled samples.

Finally, Fig. 3d shows the performance comparison when the methods are applied to the Wine Data Set [17]. 40 samples are randomly chosen to compose the labeled subset. When all the samples in the labeled subset are correctly labeled, the proposed algorithm already performs a little better than the others. As the mislabeled subset increases, this difference becomes more notable because the classification error rates of the other algorithms increases more than the classification error rate of the proposed method. With up to 30% of mislabeled samples in the labeled subset, the proposed algorithm seems not to be affected at all by the wrong labels as it keeps a stable performance. With 35% of mislabeled samples and beyond, there is a decrease in the proposed method performance. However, it still performs much better than LGC, LP, and LNP with up to 55% of mislabeled samples. When 50% of the labeled subset is composed by mislabeled samples, the proposed method still has less than half of the amount of classification mistakes made by LGC, LP and LNP.

IV. Conclusions

This paper proposes a nature-inspired method for semi-supervised classification using teams of walking particles competing for network nodes, where each team corresponds to one of the classes label. It is specifically designed to address the problem of learning from imperfect data, i.e., data sets which may have mislabeled samples. Nodes that are initially assigned to the wrong class (label) are usually more susceptible to attacks from the team of particles which represents its correct label. Therefore, these nodes can be taken by the correct team after a sufficient amount of movements, stopping the wrong label spreading.

Computer simulations results indicates that the proposed model is robust to the presence of mislabeled data. They also indicate the presence of critical points in the performance curve as the mislabeled samples subset grows, and how these critical points are related to the network size and average node degree. These results may be useful for designing secure and robust machine learning techniques. In the comparison against other representative graph-based semi-supervised methods, the proposed algorithm performed significatively better than those when applied to some artificial and real-world data sets with increasing percentage of mislabeled samples.

References

[1] X. Zhu, "Semi-supervised learning literature survey," Computer Sciences, University of Wisconsin-Madison, Tech. Rep. 1530, 2005.

[2] O. Chapelle, B. Schölkopf, and A. Zien, Eds., *Semi-Supervised Learning*, ser. Adaptive Computation and Machine Learning. Cambridge, MA: The MIT Press, 2006.

[3] S. Abney, *Semisupervised Learning for Computational Linguistics*. CRC Press, 2008.

[4] D. K. Slonim, "Learning from imperfect data in theory and practice," Cambridge, MA, USA, Tech. Rep., 1996.

[5] T. Krishnan, "Efficiency of learning with imperfect supervision," *Pattern Recogn.*, vol. 21, no. 2, pp. 183–188, 1988.

[6] P. Hartono and S. Hashimoto, "Learning from imperfect data," *Appl. Soft Comput.*, vol. 7, no. 1, pp. 353–363, 2007.

[7] M.-R. Amini and P. Gallinari, "Semi-supervised learning with an imperfect supervisor," *Knowl. Inf. Syst.*, vol. 8, no. 4, pp. 385–413, 2005.

[8] ——, "Semi-supervised learning with explicit misclassification modeling," in *IJCAI'03: Proceedings of the 18th international joint conference on Artificial intelligence*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2003, pp. 555–560.

[9] M. G. Quiles, L. Zhao, R. L. Alonso, and R. A. F. Romero, "Particle competition for complex network community detection," *Chaos*, vol. 18, no. 3, p. 033107, 2008.

[10] F. A. Breve, L. Zhao, M. G. Quiles, W. Pedrycz, and J. Liu, "Particle competition and cooperation in networks for semi-supervised learning," *IEEE Transactions on Knowledge and Data Engineering*, no. PrePrints - DOI: 10.1109/TKDE.2011.119, 2011.

[11] F. A. Breve, L. Zhao, and M. G. Quiles, "Semi-supervised learning from imperfect data through particle cooperation and competition," in *Neural Networks (IJCNN), The 2010 International Joint Conference on*, july 2010, pp. 1 –8.

[12] L. Danon, A. Díaz-Guilera, J. Duch, and A. Arenas, "Comparing community structure identification," *Journal of Statistical Mechanics: Theory and Experiment*, vol. 9, p. P09008, 2005.

[13] D. Zhou, O. Bousquet, T. N. Lal, J. Weston, and B. Schölkopf, "Learning with local and global consistency," in *Advances in Neural Information Processing Systems*, vol. 16. MIT Press, 2004, pp. 321–328.

[14] X. Zhu and Z. Ghahramani, "Learning from labeled and unlabeled data with label propagation," Carnegie Mellon University, Pittsburgh, Tech. Rep. CMU-CALD-02-107, 2002.

[15] F. Wang and C. Zhang, "Label propagation through linear neighborhoods," *IEEE Transactions on Knowledge and Data Engineering*, vol. 20, no. 1, pp. 55–67, Jan. 2008.

[16] R. Duin, P. Juszczak, P. Paclik, E. Pekalska, D. de Ridder, D. Tax, and S. Verzakov, "Prtools4.1, a matlab toolbox for pattern recognition," Delft University of Technology, 2007. [Online]. Available: http://prtools.org

[17] A. Frank and A. Asuncion, "UCI machine learning repository," 2010. [Online]. Available: http://archive.ics.uci.edu/ml